6.3. Recursively Enumerable Sets

Consider the set

$$A = \{x \in \mathbb{N} \mid \varphi_x(a) \text{ is defined}\},$$

where $a \in \mathbb{N}$ is any fixed natural number.

By Rice's Theorem, A is not recursive (check this).

We claim that A is the range of a recursive function g. For this, we use the T-predicate.

We produce a function which is actually primitive recursive.

First, note that A is nonempty (why?), and let x_0 be any index in A.

We define g by primitive recursion as follows:

$$g(0) = x_0,$$
$$g(x+1) = \begin{cases} \Pi_1(x) & \text{if } T(\Pi_1(x), a, \Pi_2(x)), \\ x_0 & \text{otherwise.} \end{cases}$$

Since this type of argument is new, it is helpful to explain informally what g does.

For every input x, the function g tries finitely many steps of a computation on input a of some partial recursive function.

The computation is given by $\Pi_2(x)$, and the partial function is given by $\Pi_1(x)$.

Since Π_1 and Π_2 are projection functions, when x ranges over $\mathbb{N}$, both $\Pi_1(x)$ and $\Pi_2(x)$ also range over $\mathbb{N}$.

Such a process is called a *dovetailing* computation.

Therefore all computations on input a for all partial recursive functions will be tried, and the indices of the partial recursive functions converging on input a will be selected.

Definition 6.3.1 A subset X of $\mathbb{N}$ is *recursively enumerable* iff either $X = \emptyset$, or X is the range of some total recursive function. Similarly, a subset X of Σ^* is *recursively enumerable* iff either $X = \emptyset$, or X is the range of some total recursive function.

Remark: It should be noted that the definition of an *r.e set* given in Definition 6.3.1 is *different* from the earlier Definition 4.8.1 given in terms of acceptance by a Turing machine. The equivalence of these two definitions will be proved in Lemma 6.3.3.

For short, a *recursively enumerable set* is also called an *r.e. set*.

The following Lemma relates recursive sets and recursively enumerable sets:

Lemma 6.3.2 *A set A is recursive iff both A and its complement $\overline{A}$ are recursively enumerable.*

Proof. Assume that A is recursive. Then, it is trivial that its complement is also recursive.

Hence, we only have to show that a recursive set is recursively enumerable.

The empty set is recursively enumerable by definition. Otherwise, let $y \in A$ be any element. Then, the function f defined such that

$$f(x) = \begin{cases} x & \text{iff } C_A(x) = 1, \\ y & \text{iff } C_A(x) = 0, \end{cases}$$

for all $x \in \mathbb{N}$ is recursive and has range A.

Conversely, assume that both A and $\overline{A}$ are recursively enumerable.

If either A or $\overline{A}$ is empty, then A is recursive.

Otherwise, let $A = f(\mathbb{N})$ and $\overline{A} = g(\mathbb{N})$, for some recursive functions f and g.

We define the function C_A as follows:

$$C_A(x) = \begin{cases} 1 & \text{if } f(\min y[f(y) = x \;\vee\; g(y) = x]) = x, \\ 0 & \text{otherwise.} \end{cases}$$

The function C_A lists A and $\overline{A}$ in parallel, waiting to see whether x turns up in A or in $\overline{A}$.

Note that x must eventually turn up either in A or in $\overline{A}$, so that C_A is a total recursive function. $\square$

Our next goal is to show that the recursively enumerable sets can be given several equivalent definitions.

Lemma 6.3.3 *For any subset A of $\mathbb{N}$, the following properties are equivalent:*

(1) A is empty or A is the range of a primitive recursive function (Rosser, 1936).

(2) A is recursively enumerable.

(3) A is the range of a partial recursive function.

(4) A is the domain of a partial recursive function.

Note that (4) is equivalent to Definition 4.8.1 (in terms of Turing machines).

More intuitive proofs of the implications (3) $\Rightarrow$ (4) and (4) $\Rightarrow$ (1) can be given.

Assume that $A \neq \emptyset$ and that $A = range(g)$, where g is a partial recursive function.

Assume that g is computed by a RAM program P.

To compute $f(x)$, we start computing the sequence

$$g(0), g(1), \ldots$$

looking for x. If x turns up as say $g(n)$, then we output n.

Otherwise the computation diverges. Hence, the domain of f is the range of g.

Assume now that A is the domain of some partial recursive function g, and that g is computed by some Turing machine M.

We construct another Turing machine performing the following steps:

(0) Do one step of the computation of $g(0)$

$\dots$

(n) Do $n+1$ steps of the computation of $g(0)$

Do n steps of the computation of $g(1)$

$\dots$

Do 2 steps of the computation of $g(n-1)$

Do 1 step of the computation of $g(n)$

During this process, whenever the computation of some $g(m)$ halts, we output m.

In this fashion, we will enumerate the domain of g, and since we have constructed a Turing machine that halts for every input, we have a total recursive function.

The following Lemma can easily be shown using the proof technique of Lemma 6.3.3:

Lemma 6.3.4 *The following properties hold:*

(1) There is a recursive function h such that

$$range(\varphi_x) = dom(\varphi_{h(x)})$$

for all $x \in \mathbb{N}$.

(2) There is a recursive function k such that

$$dom(\varphi_x) = range(\varphi_{k(x)})$$

and $\varphi_{k(x)}$ is total recursive, for all $x \in \mathbb{N}$ such that $dom(\varphi_x) \neq \emptyset$.

Using Lemma 6.3.3, we can prove that K is an r.e. set. Indeed, we have $K = dom(f)$, where

$$f(x) = \varphi_{univ}(x, x)$$

for all $x \in \mathbb{N}$.

The set

$$K_0 = \{\langle x, y\rangle \mid \varphi_x(y) \text{ converges}\}$$

is also an r.e. set, since $K_0 = dom(g)$, where

$$g(z) = \varphi_{univ}(\Pi_1(z), \Pi_2(z)),$$

which is partial recursive.

The sets K and K_0 are examples of r.e. sets that are not recursive.

We can now prove that there are sets that are not r.e.

Lemma 6.3.5 *For any indexing of the partial recursive functions, the complement $\overline{K}$ of the set*

$$K = \{x \in \mathbb{N} \mid \varphi_x(x) \ \textit{converges}\}$$

is not recursively enumerable.

Proof. If $\overline{K}$ was recursively enumerable, since K is also recursively enumerable, by Lemma 6.3.2, the set K would be recursive, a contradiction. $\square$

The sets $\overline{K}$ and $\overline{K_0}$ are examples of sets that are not r.e.

This shows that the r.e. sets are not closed under complementation. However, we leave it as an exercise to prove that the r.e. sets are closed under union and intersection.

We will prove later on that TOTAL is not r.e.

This is rather unpleasant. Indeed, this means that there is no way of effectively listing all algorithms (all total recursive functions).

Hence, in a certain sense, the concept of partial recursive function (procedure) is more natural than the concept of a (total) recursive function (algorithm).

The next two Lemmas give other characterizations of the r.e. sets and of the recursive sets.

Lemma 6.3.6 *The following properties hold:*

(1) A set A is r.e. iff either it is finite or it is the range of an injective recursive function.

(2) A set A is r.e. if either it is empty or it is the range of a monotonic partial recursive function.

(3) A set A is r.e. iff there is a Turing machine M such that, for all $x \in \mathbb{N}$, M halts on x iff $x \in A$.

Lemma 6.3.7 *A set A is recursive iff either it is finite or it is the range of a strictly increasing recursive function.*

Another important result relating the concept of partial recursive function and that of an r.e set is given below.

Theorem 6.3.8 *For every unary partial function f, the following properties are equivalent:*

(1) f is partial recursive.

(2) The set

$$\{\langle x, f(x)\rangle \mid x \in dom(f)\}$$

is r.e.

Using our indexing of the partial recursive functions and Lemma 6.3.3, we obtain an indexing of the r.e sets.

Definition 6.3.9 For any acceptable indexing $\varphi_0, \varphi_1, \ldots$ of the partial recursive functions, we define the enumeration $W_0, W_1, \ldots$ of the r.e. sets by setting

$$W_x = dom(\varphi_x).$$

We now describe a technique for showing that certain sets are r.e but not recursive, or complements of r.e. sets that are not recursive, or not r.e, or neither r.e. nor the complement of an r.e. set. This technique is known as *reducibility*.

6.4. Reducibility and Complete Sets

We already used the notion of reducibility in the proof of Lemma 6.2.5 to show that TOTAL is not recursive.

Definition 6.4.1 A set A is *many-one reducible* to a set B if there is a total recursive function f such that

$$x \in A \quad \text{iff} \quad f(x) \in B$$

for all $x \in A$. We write $A \leq B$, and for short, we say that A is *reducible* to B.

Lemma 6.4.2 *Let A, B, C be subsets of $\mathbb{N}$ (or Σ^*). The following properties hold:*

(1) If $A \leq B$ and $B \leq C$, then $A \leq C$.

(2) If $A \leq B$ then $\overline{A} \leq \overline{B}$.

(3) If $A \leq B$ and B is r.e., then A is r.e.

(4) If $A \leq B$ and A is not r.e., then B is not r.e.

(5) If $A \leq B$ and B is recursive, then A is recursive.

(6) If $A \leq B$ and A is not recursive, then B is not recursive.

Another important concept is the concept of a complete set.

Definition 6.4.3 An r.e. set A is *complete w.r.t. many-one reducibility* iff every r.e. set B is reducible to A, i.e., $B \leq A$.

For simplicity, we will often say *complete* for *complete w.r.t. many-one reducibility*.

Theorem 6.4.4 *The following properties hold:*

(1) If A is complete, B is r.e., and $A \leq B$, then B is complete.

(2) K_0 is complete.

(3) K_0 is reducible to K.

As a corollary of Theorem 6.4.4, the set K is also complete.

Definition 6.4.5 Two sets A and B have the same *degree of unsolvability* or are *equivalent* iff $A \leq B$ and $B \leq A$.

Since K and K_0 are both complete, they have the same degree of unsolvability.

We will now investigate the reducibility and equivalence of various sets. Recall that

$$\mathrm{TOTAL} = \{x \in \mathbb{N} \mid \varphi_x \text{ is total}\}.$$

We define EMPTY and FINITE, as follows:

$$\mathrm{EMPTY} = \{x \in \mathbb{N} \mid \varphi_x \text{ is undefined for all input}\},$$
$$\mathrm{FINITE} = \{x \in \mathbb{N} \mid \varphi_x \text{ has a finite domain}\}.$$

Then,

$$\overline{\text{FINITE}} = \{x \in \mathbb{N} \mid \varphi_x \text{ has an infinite domain}\},$$

so that,

EMPTY $\subset$ FINITE and TOTAL $\subset \overline{\text{FINITE}}$.

Lemma 6.4.6 *We have* $K_0 \leq \overline{\text{EMPTY}}$.

Lemma 6.4.7 *The following properties hold:*

(1) EMPTY *is not r.e.*

(2) $\overline{\text{EMPTY}}$ *is r.e.*

(3) $\overline{K}$ *and* EMPTY *are equivalent.*

(4) $\overline{\text{EMPTY}}$ *is complete.*

Lemma 6.4.8 *The following properties hold:*

(1) TOTAL *and* $\overline{\text{TOTAL}}$ *are not r.e.*

(2) FINITE *and* $\overline{\text{FINITE}}$ *are not r.e.*

From Lemma 6.4.8, we have TOTAL $\leq \overline{\text{FINITE}}$. It turns out that $\overline{\text{FINITE}} \leq$ TOTAL, and TOTAL and $\overline{\text{FINITE}}$ are equivalent.

Lemma 6.4.9 *The sets* TOTAL *and* $\overline{\text{FINITE}}$ *are equivalent.*

We now turn to the recursion Theorem.